



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

09/975,945

10/11/2001

Charles Paclat

B241 1780.3

9612

74739

7590

04/14/2009

Womble Carlyle Sandridge & Rice, PLLC

Oracle International Corporation

Attn: Patent Docketing 32nd Floor

Post Office Box 7037

Atlanta, GA 30357-0037

EXAMINER

CHEN, QING

ART UNIT

PAPER NUMBER

2191

MAIL DATE

DELIVERY MODE

04/14/2009

PAPER

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

<b>Office Action Summary</b>	<b>Application No.</b> 09/975,945	<b>Applicant(s)</b> PACLAT, CHARLES	
	<b>Examiner</b> Qing Chen	<b>Art Unit</b> 2191	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

### Status

- 1) ☒ Responsive to communication(s) filed on 27 January 2009.
- 2a) ☒ This action is **FINAL**.                      2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

### Disposition of Claims

- 4) ☒ Claim(s) 1-3 and 6-24 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-3 and 6-24 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

### Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

### Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All    b) ☐ Some \*    c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

### Attachment(s)

- |  |   |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892)                     | 4) <input type="checkbox"/> Interview Summary (PTO-413)           |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____                                      |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)          | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date _____  | 6) <input type="checkbox"/> Other: _____                          |

**DETAILED ACTION**

1. This Office action is in response to the amendment filed on January 27, 2009.
2. **Claims 1-3 and 6-24** are pending.
3. **Claims 1, 3, 6-12, and 14-18** have been amended.
4. **Claims 4 and 5** have been canceled.
5. **Claims 19-24** have been added.
6. The objection to the title is withdrawn in view of Applicant's amendments to the title.
7. The objection to the abstract is withdrawn in view of Applicant's amendments to the abstract.
8. Applicant has failed to address the objection to the specification due to the use of trademarks. Accordingly, this objection is maintained and further explained hereinafter.
9. The objections to Claims 6, 8, and 18 are withdrawn in view of Applicant's amendments to the claims.
10. The 35 U.S.C. § 112, second paragraph, rejections of Claims 1-11 and 13-18 are withdrawn in view of Applicant's amendments to the claims. However, Applicant's amendments to the claims fail to fully address the rejection of Claim 12 due to insufficient antecedent basis. Accordingly, this rejection is maintained and further explained hereinafter.
11. The 35 U.S.C. § 101 rejections of Claims 1-18 are withdrawn in view of Applicant's amendments to the claims.

***Response to Amendment***

***Specification***

12. The use of trademarks, such as EJB and JAVA, has been noted in this application. Trademarks should be capitalized wherever they appear (capitalize each letter OR accompany each trademark with an appropriate designation symbol, *e.g.*, <sup>TM</sup> or ®) and be accompanied by the generic terminology (use trademarks as adjectives modifying a descriptive noun, *e.g.*, “the JAVA programming language”).

Although the use of trademarks is permissible in patent applications, the proprietary nature of the marks should be respected and every effort made to prevent their use in any manner, which might adversely affect their validity as trademarks.

***Claim Objections***

13. **Claims 2, 3, and 6-24** are objected to because of the following informalities:
- **Claims 2, 3, and 6-18** recite the category of invention “[t]he method.” Applicant is advised to change this category of invention to read “[t]he computer-implemented method” for the purpose of providing it with proper explicit antecedent basis.
  - **Claims 19-24** contain a typographical error: “A/The computer readable media” should read -- A/The computer readable medium --.
- Appropriate correction is required.

***Claim Rejections - 35 USC § 112***

14. The following is a quotation of the second paragraph of 35 U.S.C. 112:

Art Unit: 2191

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

15. **Claim 12** is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

**Claim 12** recites the limitation “said sub-domains.” There is insufficient antecedent basis for this limitation in the claim. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading “said one or more sub-domains” for the purpose of further examination.

***Claim Rejections - 35 USC § 103***

16. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

17. **Claims 1-3, 6-15, and 19-24** are rejected under 35 U.S.C. 103(a) as being unpatentable over **US 6,237,135 (hereinafter “Timbol”)** in view of **US 6,167,564 (hereinafter “Fontana”)**.

As per **Claim 1**, Timbol discloses:

- (a) analyzing a business domain to determine functional requirements of said business domain (*see Column 9: 54-63, “Since the component is used within a development environment,*

Art Unit: 2191

*"bean info" (i.e., information) is provided as a meta-data companion to a Java bean. For instance, if one had an "account balance" Java bean, there would also exist an "account balance" Java bean info class."; Column 10: 50-52, "The user employs the Java Bean Wizard 300 to specify the name of the bean, the package it will be in, and the class it extends from.";*

- (b) transforming said functional requirements into an object oriented component model, wherein said functional requirements include a data model and a process model of said business domain, and the object oriented component model encapsulates the data model and process model (*see Column 8: 61-67, "The component palette 264 displays components available in the JBuilder component library. Components are the elements which a user employs to build his or her applications. They include all of the visible parts of an application, such as dialog boxes and buttons, as well as those which are not visible while the application is running (e.g., system timers)."; Column 9: 38-41, "It provides a defined model of how a reusable component in Java should be packaged, so that the component could be freely used in any Java development environment." and 51-53, "In order to support this level of functionality, the Sun Java Bean model specifies a set of design patterns of how component should be coded (i.e., structured in source code)."; Column 10: 27-33, "A Java Bean can be a discrete component used in building a user interface, or a non-UI component such as a data module or computation engine. At its simplest, a Java Bean is a public Java class that has a constructor with no parameters. Java Beans usually have properties, methods, and events that follow certain naming conventions (also known as "design patterns")."; and*

- (c) building said reusable component in accordance with said object oriented component model that encompasses a business functionality of said business domain (*see*

Art Unit: 2191

*Column 19: 15 and 16, "The system provides an Enterprise Java Bean Wizard for creating Enterprise Java Beans.").*

However, Timbol does not disclose:

- wherein the functional requirements comprise a list of inputs for said business domain.

Fontana discloses

- wherein the functional requirements comprise a list of inputs for said business domain  
*(see Column 7: 38-47, "A typical business domain generally comprises a wide range of functionalities, which in aggregation form the overall functions of a business domain. A clearly defined coherent description of such functionalities are called business models." and "A Business Model includes descriptions of people's roles, processes and procedures, and business rules.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Fontana into the teaching of Timbol to include wherein the functional requirements comprise a list of inputs for said business domain. The modification would be obvious because one of ordinary skill in the art would be motivated to model the overall functions of the business domain.

As per **Claim 2**, the rejection of **Claim 1** is incorporated; and Timbol further discloses:

- modifying said functional requirements by a user *(see Column 11: 43-45, "As described below, however, the system provides visual designers and additional methodology for allowing the user to further customize the bean." ); and*

Art Unit: 2191

- repeating the steps (b) and (c) to provide a parallel development process (*see Column 10: 17-21, "Further, once the user has created a "Java Bean" (i.e., component), he or she can continue to use the BeansExpress visual designers and methodology as true "two-way" tools to make further changes to the generated component, as needed."*).

As per **Claim 3**, the rejection of **Claim 1** is incorporated; and Timbol further discloses:

- wherein said reusable component is extensible and configurable (*see Column 10: 34-36, "Like other types of components, Java Beans are reusable pieces of code that can be updated with minimal impact on the testing of the program they become a part of."*).

As per **Claim 6**, the rejection of **Claim 1** is incorporated; however, Timbol does not disclose:

- wherein the step of analyzing includes the step of generating the list of inputs, each input identifying a resource that relates to said business domain.

Fontana discloses:

- wherein the step of analyzing includes the step of generating the list of inputs, each input identifying a resource that relates to said business domain (*see Column 7: 38-47, "A typical business domain generally comprises a wide range of functionalities, which in aggregation form the overall functions of a business domain. A clearly defined coherent description of such functionalities are called business models." and "A Business Model includes descriptions of people's roles, processes and procedures, and business rules."*).



Art Unit: 2191

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Fontana into the teaching of Timbol to include wherein the step of analyzing includes the step of generating the list of inputs, each input identifying a resource that relates to said business domain. The modification would be obvious because one of ordinary skill in the art would be motivated to analyze the overall functions of the business domain.

As per **Claim 7**, the rejection of **Claim 6** is incorporated; however, Timbol and Fontana do not disclose:

- generating an eFunction matrix from said list of inputs.

Official Notice is taken that it is old and well-known within the computing art to transform information from a list format to a matrix or table format. A matrix or table is commonly utilized to easily compare and contrast related information. Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to include generating an eFunction matrix from said list of inputs. The modification would be obvious because one of ordinary skill in the art would be motivated to easily compare and contrast related information.

As per **Claim 8**, the rejection of **Claim 1** is incorporated; however, Timbol does not disclose:

- wherein the step of transforming transforms said functional requirements using a unified modeling language (UML) tool to generate said object oriented component model.

Art Unit: 2191

Fontana discloses:

- wherein the step of transforming transforms said functional requirements using a unified modeling language (UML) tool to generate said object oriented component model (*see Column 8: 54-57, "Included within the repository 32 is a Business Model module 66. As noted, the module 66 may be written in UML with extensions, which will be amplified hereinafter."*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Fontana into the teaching of Timbol to include wherein the step of transforming transforms said functional requirements using a unified modeling language (UML) tool to generate said object oriented component model. The modification would be obvious because one of ordinary skill in the art would be motivated to utilize a de-facto industry standard for object-oriented analysis and design (*see Fontana - Column 6: 66 and 67 to Column 7: 1 and 2*).

As per **Claim 9**, the rejection of **Claim 8** is incorporated; and Timbol further discloses:

- wherein said object oriented component model includes a plurality of classes (*see Column 10: 24-27, "A Java Bean is a collection of one or more Java classes, often bundled into a single JAR (Java Archive) file, that serves as a self-contained, reusable component."*).

As per **Claim 10**, the rejection of **Claim 9** is incorporated; however, Timbol does not disclose:

Art Unit: 2191

- wherein the step of building builds said reusable component from at least one of the following class stereotypes: Belonging, Session, Entity, Configurable Entity, Business Policy and Workflow.

Fontana discloses:

- wherein the step of building builds said reusable component from at least one of the following class stereotypes: Belonging, Session, Entity, Configurable Entity, Business Policy and Workflow (*see Column 8: 54-57, "Included within the repository 32 is a Business Model module 66. As noted, the module 66 may be written in UML with extensions, which will be amplified hereinafter."*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Fontana into the teaching of Timbol to include wherein the step of building builds said reusable component from at least one of the following class stereotypes: Belonging, Session, Entity, Configurable Entity, Business Policy and Workflow. The modification would be obvious because one of ordinary skill in the art would be motivated to utilize a de-facto industry standard for object-oriented analysis and design (*see Fontana - Column 6: 66 and 67 to Column 7: 1 and 2*).

As per **Claim 11**, the rejection of **Claim 1** is incorporated; however, Timbol does not disclose:

- wherein the step of transforming includes the step of mapping eXtensible Markup Language (XML) to said object oriented component model.

Fontana discloses:

Art Unit: 2191

- wherein the step of transforming includes the step of mapping eXtensible Markup Language (XML) to said object oriented component model (*see Column 6: 63-66, "The XML component 40 is linked to two models within the repository 32. The first is a relational database ("RDB") model 43 and the second is a Unified Modeling Language ("UML") model 44."*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Fontana into the teaching of Timbol to include wherein the step of transforming includes the step of mapping eXtensible Markup Language (XML) to said object oriented component model. The modification would be obvious because one of ordinary skill in the art would be motivated to exchange messages in the proper format (*see Fontana - Column 6: 61 and 62*).

As per **Claim 12**, the rejection of **Claim 1** is incorporated; however, Timbol does not disclose:

- wherein the step of analyzing includes the step of dividing said business domain into one or more sub-domains and determining functional requirements for each of said one or more sub-domains; and wherein the step of transforming transforms each of said functional requirements for said one or more sub-domains into said object oriented component model.

Fontana discloses:

- wherein the step of analyzing includes the step of dividing said business domain into one or more sub-domains and determining functional requirements for each of said one or more sub-domains; and wherein the step of transforming transforms each of said functional requirements for said one or more sub-domains into said object oriented component model (*see*

Art Unit: 2191

*Column 7: 48-55, "A business asset is defined as a particular aspect of a business, such as workflow, rules, components, transaction, database, people, strategy, laws, etc. Depending on whether an asset is independent of or dependent upon technology, they are classified as Technology Dependent and Technology Independent assets. Examples of Technology Independent assets are people and strategy while that of Technology Dependent assets are databases and workflow.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Fontana into the teaching of Timbol to include wherein the step of analyzing includes the step of dividing said business domain into one or more sub-domains and determining functional requirements for each of said one or more sub-domains; and wherein the step of transforming transforms each of said functional requirements for said one or more sub-domains into said object oriented component model. The modification would be obvious because one of ordinary skill in the art would be motivated to understand the scope of the business model (*see Fontana - Column 7: 31-33*).

As per **Claim 13**, the rejection of **Claim 1** is incorporated; and Timbol further discloses:

- wherein the step of building includes the step of generating relational mappings and deployment descriptors (*see Column 10: 36-42, "Java Beans have some unique advantages over other components, however. They are pure Java, cross-platform components. They can be installed on the IDE (e.g., JBuilder) Component Palette and used in the construction of one's program, or they can be used in other application builder tools for Java. They can be deployed in .JAR files."*).

As per **Claim 14**, the rejection of **Claim 1** is incorporated; and Timbol further discloses:

- generating end-user documentation (*see Column 8: 12-17, “Displaying documentation, such as the Help system, a BeansExpress tutorial for creating Java Bean components, the JDK API Reference, and the JBCL API Reference.”*);
- developing unit tests to test said reusable component (*see Column 8: 47-49, “Compiles the program and runs it in the Debugger using the startup parameters in the Parameters dialog box.”*); and
- generating a reference implementation of said reusable component (*see Column 8: 45, “Compiles and runs the application using the startup parameters in the Parameters dialog box.”*).

As per **Claim 15**, the rejection of **Claim 14** is incorporated; and Timbol further discloses:

- verifying said end-user documentation to said reusable component (*see Column 8: 12-17, “Displaying documentation, such as the Help system, a BeansExpress tutorial for creating Java Bean components, the JDK API Reference, and the JBCL API Reference.”*).

**Claims 19-24** are computer readable medium claims corresponding to the computer-implemented method claims above (Claims 1 and 6-10) and, therefore, are rejected for the same reasons set forth in the rejections of Claims 1 and 6-10.

Art Unit: 2191

18. **Claim 16** is rejected under 35 U.S.C. 103(a) as being unpatentable over **Timbol** in view of **Fontana** as applied to Claim 14 above, and further in view of **Matena et al., “Sun Microsystems Enterprise JavaBeans™,” March 1998 (hereinafter “Matena”)**.

As per **Claim 16**, the rejection of **Claim 14** is incorporated; however, Timbol and Fontana do not disclose:

- packaging said reusable component for deployment with container managed persistence.

Matena discloses:

- packaging said reusable component for deployment with container managed persistence (*see Page 59, “The entity component protocol allows the enterprise Bean provider either to implement the enterprise Bean’s persistence directly in the enterprise Bean class (we call this Bean-managed persistence), or delegate the enterprise Bean’s persistence to the container (we call this container-managed persistence).”*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Matena into the teaching of Timbol to include packaging said reusable component for deployment with container managed persistence. The modification would be obvious because one of ordinary skill in the art would be motivated to save the reusable component’s state (*see Matena – Page 59*).

Art Unit: 2191

19. **Claims 17 and 18** are rejected under 35 U.S.C. 103(a) as being unpatentable over **Timbol** in view of **Fontana** as applied to Claim 1 above, and further in view of “**Modeling with eBSCs**,” 1999 (hereinafter “**eBSCs**”).

As per **Claim 17**, the rejection of **Claim 1** is incorporated; however, Timbol and Fontana do not disclose:

- wherein said reusable component is a Smart component having at least one of following Smart feature: SmartKey, SmartHandle and SmartValue.

eBSCs discloses:

- wherein said reusable component is a Smart component having at least one of following Smart feature: SmartKey, SmartHandle and SmartValue (*see Page 7, “The SmartKey interface extends this functionality and requires the implementation of the Comparable interface from the java collection API. This is so that SmartKeys can be easily compared and stored in ordered lists. The result is that it is easy to model relationships that require the ordering of Entities.”*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of eBSCs into the teaching of Timbol to include wherein said reusable component is a Smart component having at least one of following Smart feature: SmartKey, SmartHandle and SmartValue. The modification would be obvious because one of ordinary skill in the art would be motivated to improve the ease of use and efficiency of the final system (*see eBSCs – Page 7*).



Art Unit: 2191

As per **Claim 18**, the rejection of **Claim 17** is incorporated; however, Timbol and Fontana do not disclose:

- wherein said Smart component is an eBusiness Smart component.

eBSCs discloses:

- wherein said Smart component is an eBusiness Smart component (*see Page 5, "A Belonging, the simplest form of eBusiness Smart Component, is a lightweight, local object that can be serialized."*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of eBSCs into the teaching of Timbol to include wherein said Smart component is an eBusiness Smart component. The modification would be obvious because one of ordinary skill in the art would be motivated to improve the ease of use and efficiency of the final system (*see eBSCs – Page 7*).

### ***Response to Arguments***

20. Applicant's arguments filed on January 27, 2009 have been fully considered, but they are not persuasive.

### ***In the Remarks, Applicant argues:***

- a) For example, the disclosure in Timbol that "the user employs the Java Bean Wizard 300 to specify the name of the bean it will be in, and the class it extends from" completely fails to anticipate the "analyzing a business domain ..." limitation of claim 1. Similarly, absolutely nothing in Timbol discloses converting a reusable component such as Enterprise Java Bean

Art Unit: 2191

component into an object oriented model such as an Enterprise Java Bean model. Timbol merely discloses the creation of a Java Bean component, not a model. The other cited prior art also fails to disclose the claimed elements

***Examiner's response:***

a) Examiner disagrees. Applicant's arguments are not persuasive for at least the following reasons:

First, with respect to the Applicant's assertion that the disclosure in Timbol that "the user employs the Java Bean Wizard 300 to specify the name of the bean it will be in, and the class it extends from" completely fails to anticipate the "analyzing a business domain ..." limitation, the Examiner respectfully submits that Timbol clearly discloses "analyzing a business domain to determine functional requirements of said business domain" (*see Column 9: 54-63, "Since the component is used within a development environment, "bean info" (i.e., information) is provided as a meta-data companion to a Java bean. For instance, if one had an "account balance" Java bean, there would also exist an "account balance" Java bean info class."*; *Column 10: 50-52, "The user employs the Java Bean Wizard 300 to specify the name of the bean, the package it will be in, and the class it extends from."*). Note that the Java Bean Wizard is used by the user to specify functional information about an "account balance" Java Bean (*e.g., name, class, etc.*). Thus, one of ordinary skill in the art would readily comprehend that the functional information about the "account balance" Java Bean are determined by analyzing the "account balance" Java Bean info class of the banking business domain.

Art Unit: 2191

Second, with respect to the Applicant's assertion that absolutely nothing in Timbol discloses converting a reusable component such as Enterprise Java Bean component into an object oriented model such as an Enterprise Java Bean model, the Examiner respectfully submits that Timbol clearly discloses "transforming said functional requirements into an object oriented component model, wherein said functional requirements include a data model and a process model of said business domain, and the object oriented component model encapsulates the data model and process model" (*see Column 8: 61-67, "The component palette 264 displays components available in the JBuilder component library. Components are the elements which a user employs to build his or her applications. They include all of the visible parts of an application, such as dialog boxes and buttons, as well as those which are not visible while the application is running (e.g., system timers)."; Column 9: 38-41, "It provides a defined model of how a reusable component in Java should be packaged, so that the component could be freely used in any Java development environment." and 51-53, "In order to support this level of functionality, the Sun Java Bean model specifies a set of design patterns of how component should be coded (i.e., structured in source code)."; Column 10: 27-33, "A Java Bean can be a discrete component used in building a user interface, or a non-UI component such as a data module or computation engine. At its simplest, a Java Bean is a public Java class that has a constructor with no parameters. Java Beans usually have properties, methods, and events that follow certain naming conventions (also known as "design patterns").*"). Note that Java Beans are implemented according to the Sun Java Bean models (object oriented component model) which specify a set of design patterns (encapsulates the data model and process model) of how Java Bean components should be coded. Thus, one of ordinary skill in the art would readily

Art Unit: 2191

comprehend that the Sun Java Bean models' design patterns are derived from the Java Beans' properties, methods, and events (transforming said functional requirements into an object oriented component model).

Therefore, for at least the reasons set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 1 and 19 are proper.

### ***Conclusion***

21. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

22. Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Qing Chen whose telephone number is 571-270-1071. The

Art Unit: 2191

Examiner can normally be reached on Monday through Thursday from 7:30 AM to 4:00 PM.

The Examiner can also be reached on alternate Fridays.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Wei Zhen, can be reached on 571-272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Q. C./

Examiner, Art Unit 2191

/Wei Y Zhen/

Supervisory Patent Examiner, Art Unit 2191